



**HOCHSCHULE
MITTWEIDA**
University of Applied Sciences

Grundlagen Digitale Forensik Computersysteme und Datenträgertechnik

Prof. Dr. rer. nat. Dirk Labudde



Bundeskriminalamt

Agenda

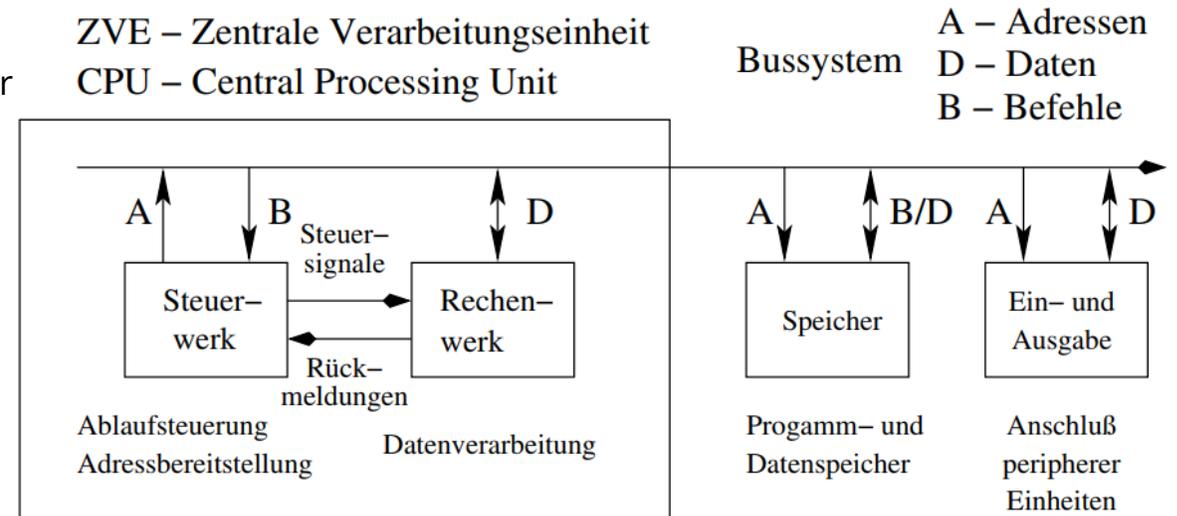
1. Grundlegender Aufbau eines Computers
 - Grundlegender Aufbau
 - CPU, Kerne, Hyperthreads
 - Speichertypen
 - Bussystem und Arbeitsspeicher
 - Ablauf des Bootvorgangs
2. Datenträgertechnik
 - Festplatten
 - Flashspeicher
 - Partitionierung (MBR, GPT)
3. RAID-Systeme

Grundlegender Aufbau eines Computers

Grundlegender Aufbau eines Computers

Grundprinzipien:

- Rechner besteht aus Hauptspeicher, Steuereinheit, Recheneinheit, Ein- und Ausgabe-Einheiten, Langzeitspeicher
- Systembus verbindet die Einheiten
- Programmsteuerung (Rechnerstruktur ist unabhängig von konkreter Aufgabe)
- Hauptspeicher enthält sowohl Daten als auch Programme



- ▶ nach John von Neumann benannt
- ▶ grundlegendes Modell eines Universalrechners
- ▶ arbeitet nach Master-Slave-Prinzip

Grundlegender Aufbau eines Computers

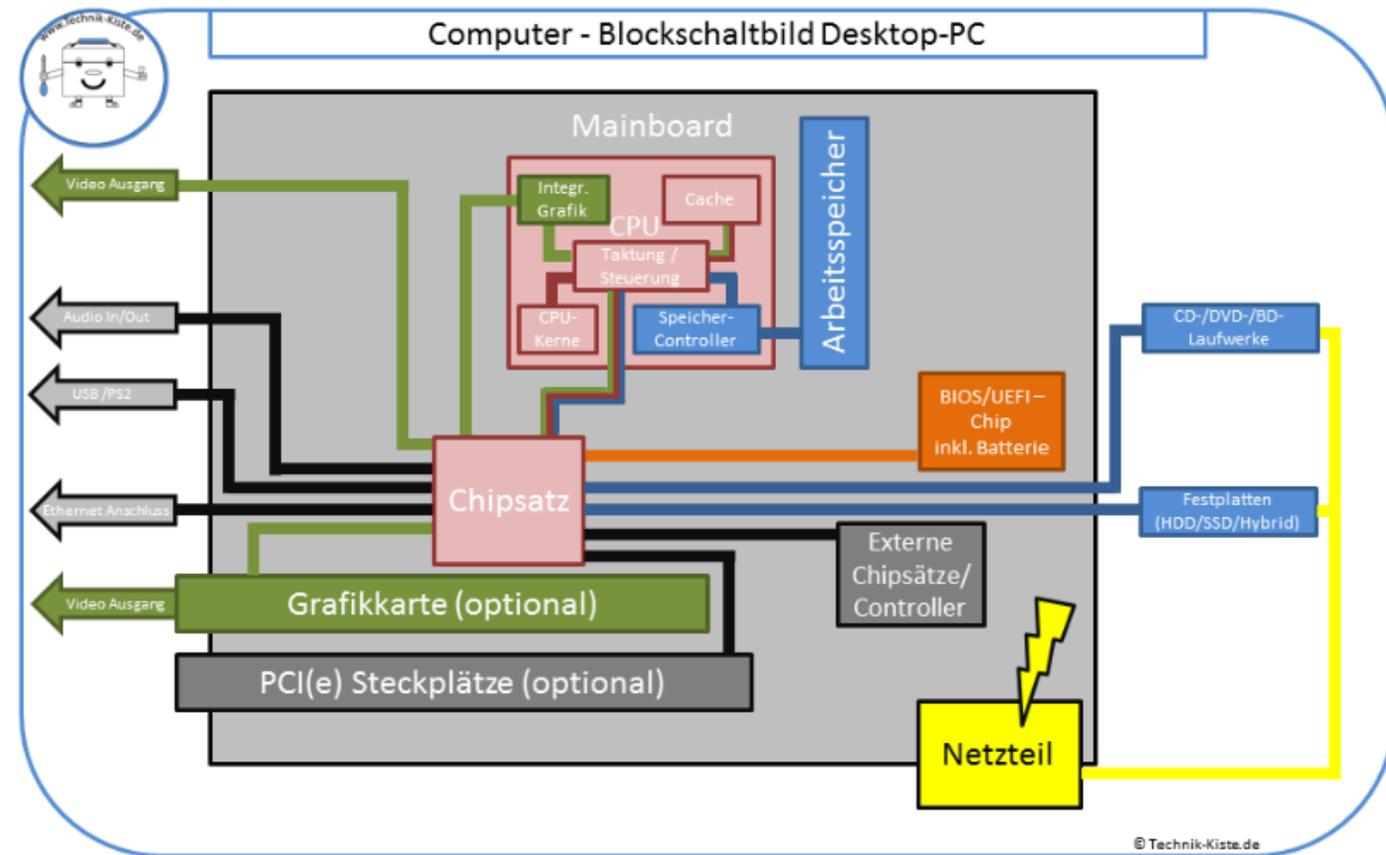
Von-Neumann-Architektur – Verarbeitungsprinzip

- Hauptspeicher besteht aus fortlaufend nummerierten Speicherplätzen. (je 1 Byte) Adresse: Nummer eines Speicherplatzes
- Programm ist eine Folge von Maschinen-Befehlen, die sequentiell (der Reihe nach) ausgeführt werden.
- Abweichung von der sequentiellen Programm-Ausführung ist durch Sprungbefehle möglich, dadurch wird die Programm-Ausführung an anderer Stelle fortgesetzt.
- Daten und Programme werden binär codiert, Zahlen werden dual dargestellt

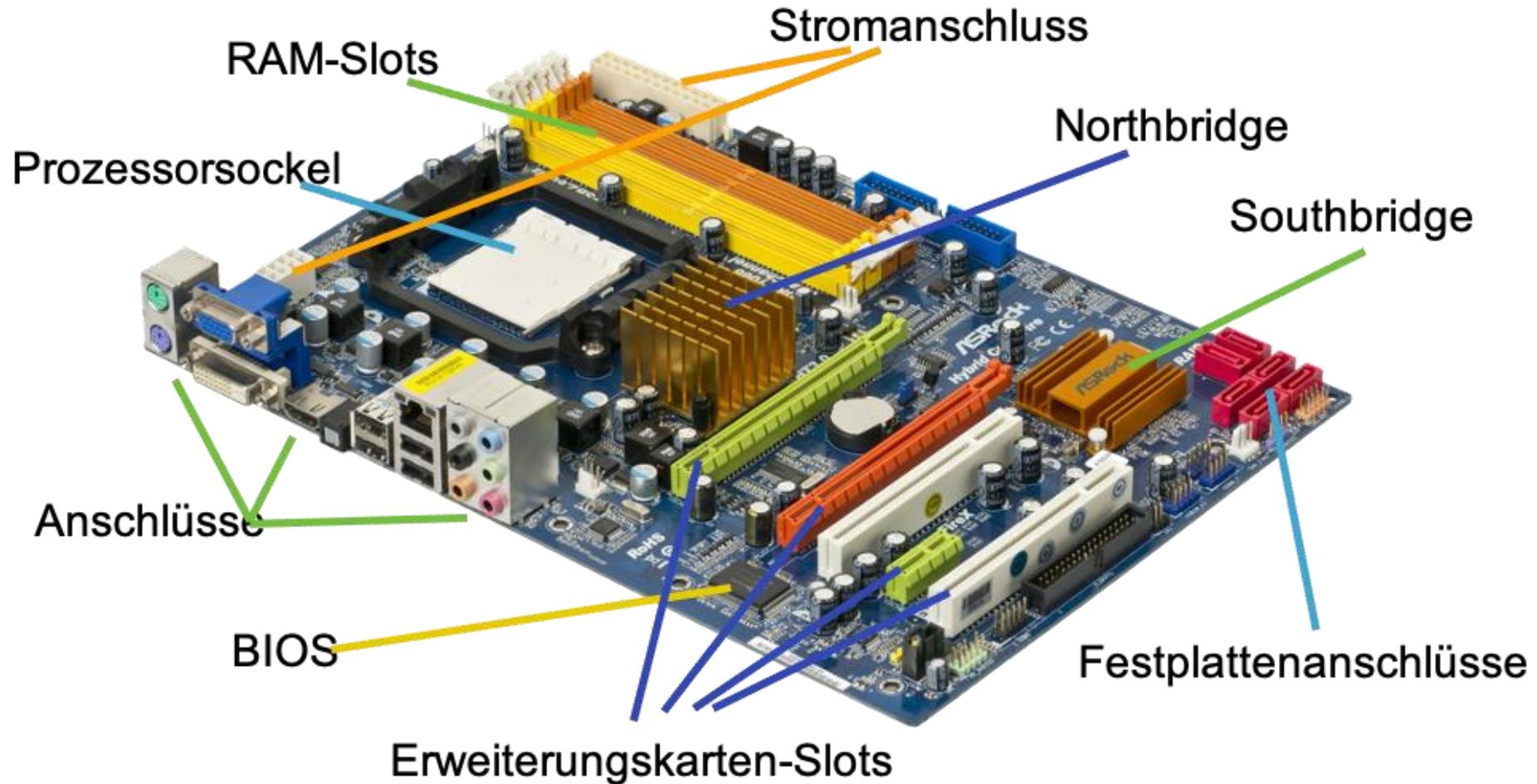


Grundlegender Aufbau eines Computers

- Prozessor (CPU – Central Processing Unit)
- Mainboard (Hauptplatine)
- Optisches Laufwerk (optional)
- Festplatte
- Hauptspeicher
- Display/Monitor
- Maus, Tastatur



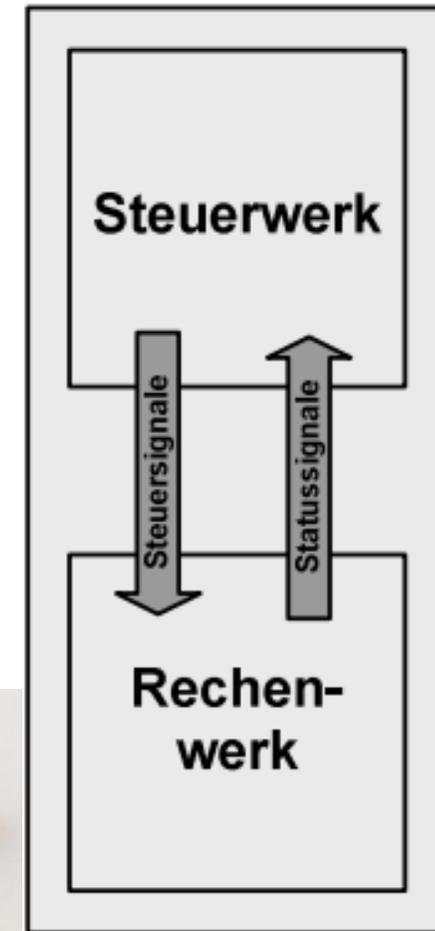
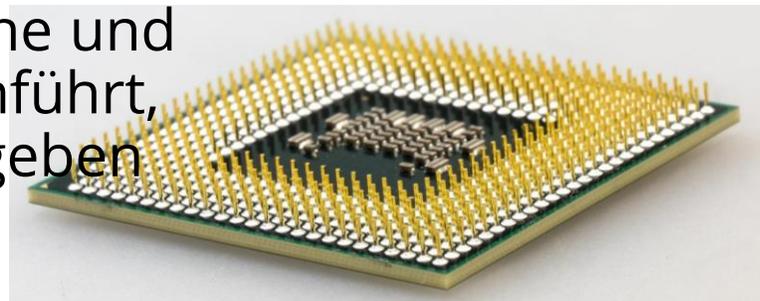
Mainboard mit Beschreibung



CPU, Kerne und Hyperthreads^[1]

Prozessor

- bearbeitet Daten, z.B. mit arithmetischen oder logischen Verknüpfungen
- regelt das Zusammenspiel aller Bauteile des Motherboards
- Steuerwerk, das alle Abläufe im System steuert
- Rechenwerk, welches verschiedenartige arithmetische und logische Operationen so durchführt, wie es vom Steuerwerk vorgegeben wird

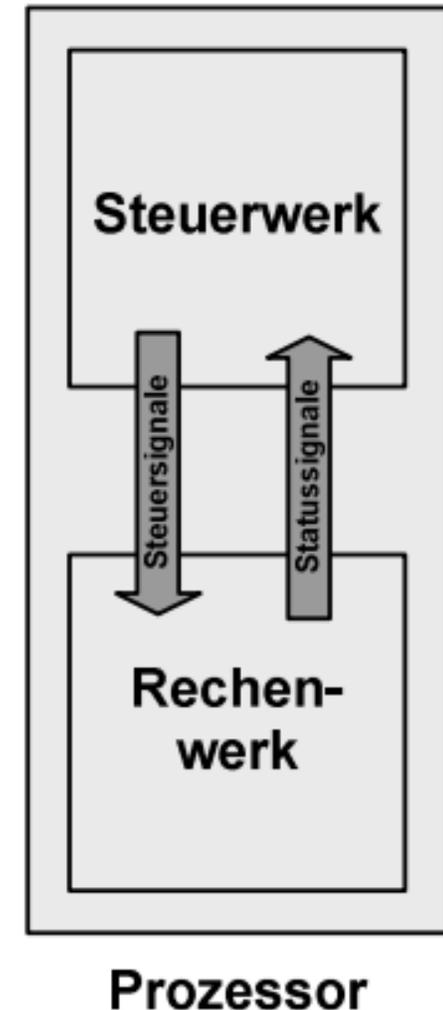


Prozessor

CPU, Kerne und Hyperthreads^[1]

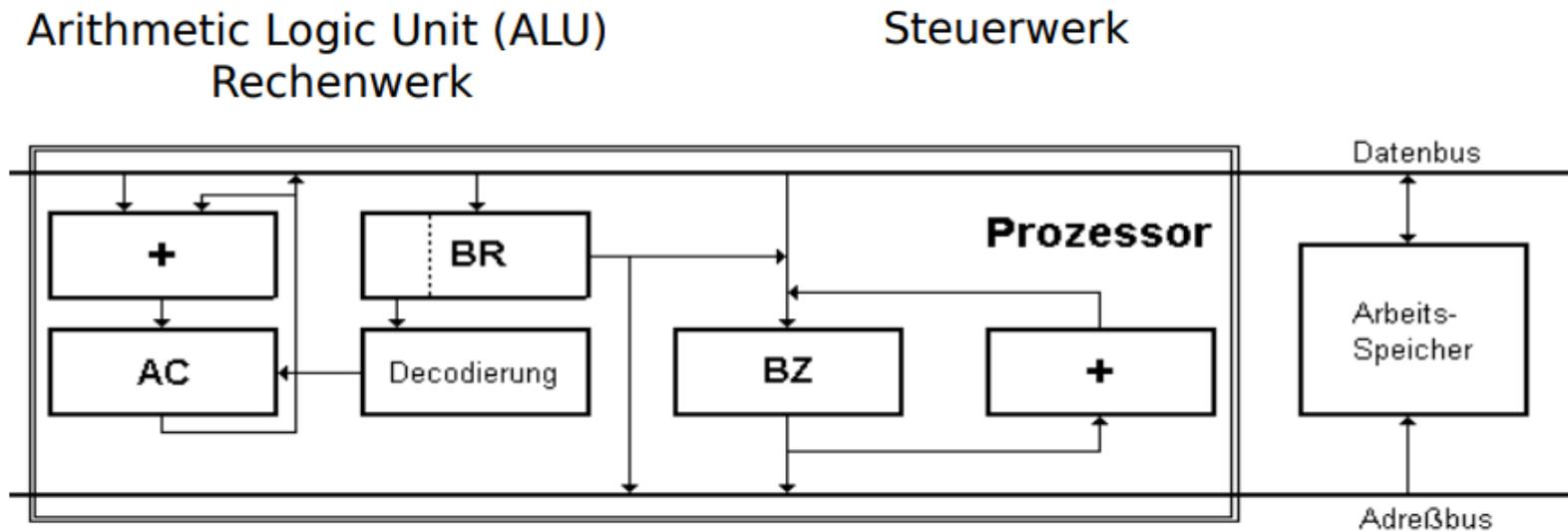
Prozessor

- Beinhaltet weitere wichtige Komponenten wie
 - Bus Interface (Busschnittstelle)
 - AGU (Address Generation Unit, Adressierungseinheit)
 - Register (Speicherstellen im Prozessor, mit denen der Prozessor besonders schnell und flexibel arbeiten kann)
- Leistungsverbesserung durch mehrere Kerne oder Hyperthreading



CPU, Kerne und Hyperthreads^[1]

Aufbau Prozessor



- AC - Akkumulator – für kumulative Additionen
- BZ - Befehlszähler
- BR Befehlsregister

CPU, Kerne und Hyperthreads^[1]

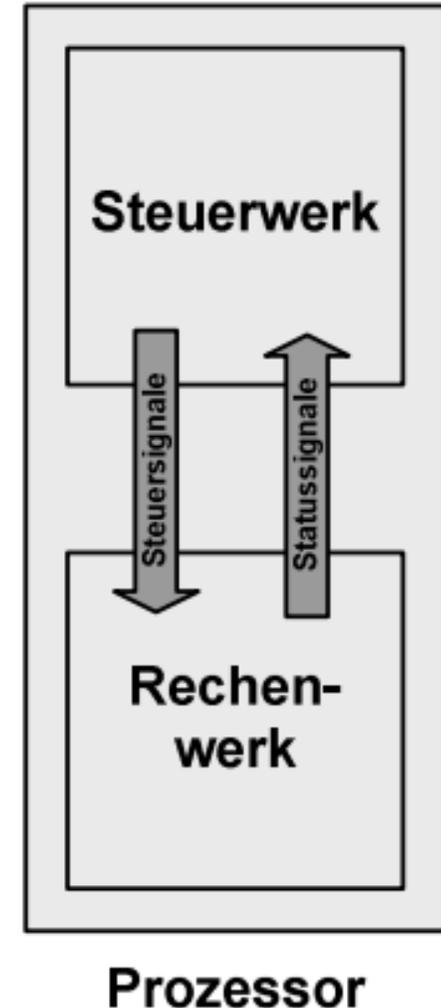
Befehlssatz eines Prozessors

- Befehlssatz = Menge seiner Maschinenbefehle
- Typische Maschinenbefehle:
 - Arithmetische Operationen (Addition, Subtraktion, Multiplikation, Division, ...)
 - Logische Operationen (AND, OR, NOT, ...)
 - Transport zwischen Speicherplätzen
 - Ein- und Ausgabeoperationen
 - Testen, Setzen, Löschen einzelner Bits in Operanden
 - Adressoperationen: Sprünge zu anderen Befehle

CPU, Kerne und Hyperthreads

Kerne

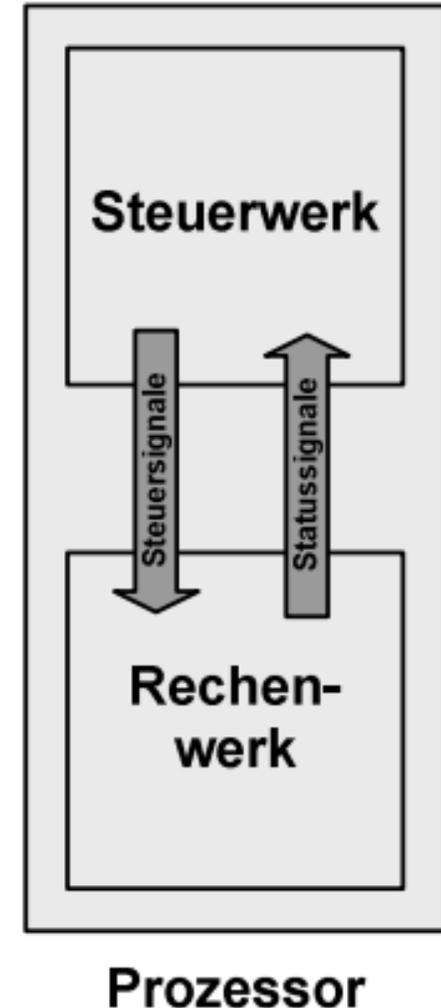
- Mehrere Kerne für Multitasking-Fähigkeit
- Zusätzliche Kerne als zusätzliche Verarbeitungseinheiten für bessere Leistung



CPU, Kerne und Hyperthreads

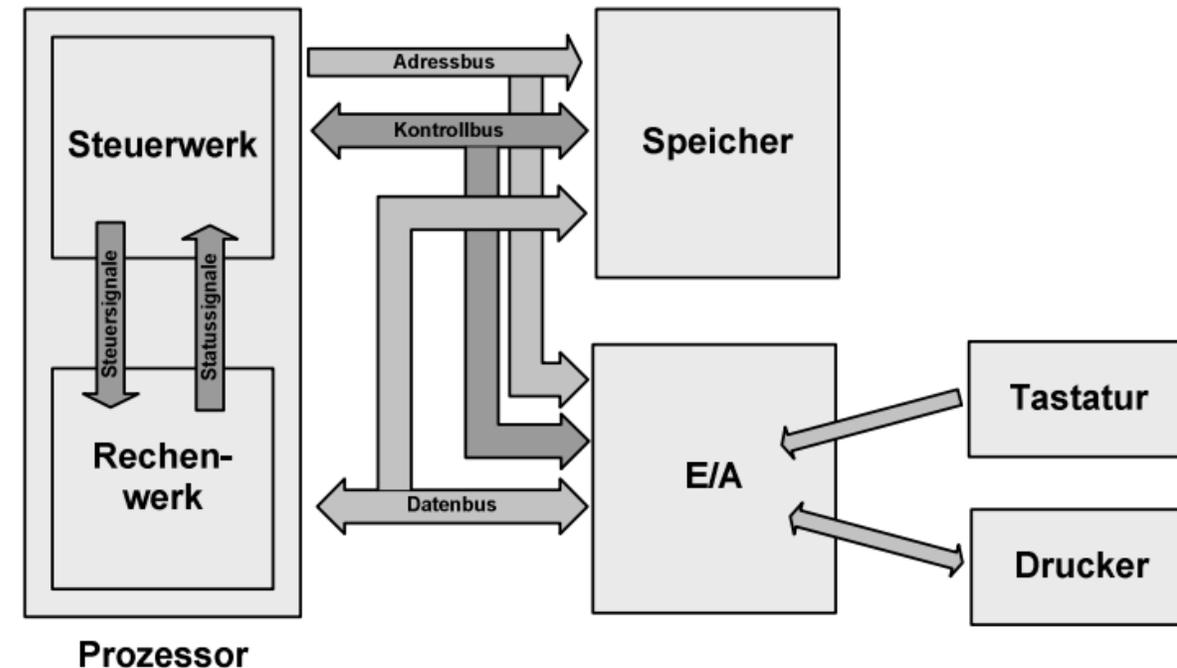
Hyperthreads

- Aufteilen des Prozessors in logische, virtuelle Kerne (nicht physisch)
 - Erhöht ebenfalls die Leistung
 - Kombination von Kernen und Hyperthreading möglich
- 8 Kerne mit aktiviertem Hyperthreading arbeiten wie 16 Kerne (virtuell)



Bussysteme und Arbeitsspeicher^[1]

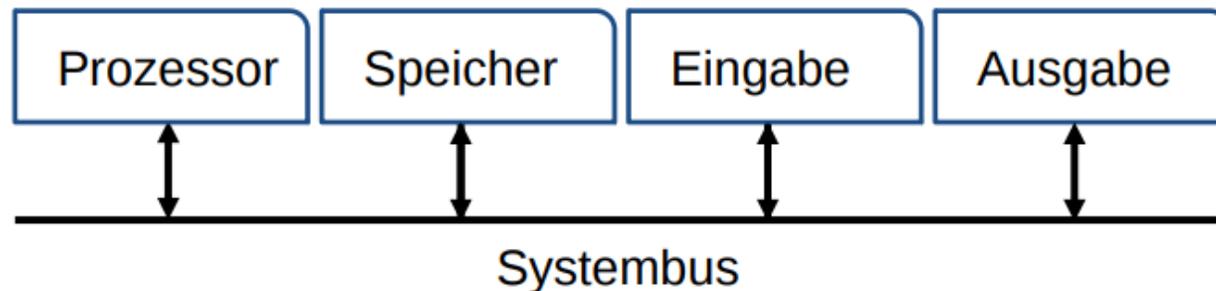
- Abbildung: Allgemeiner Aufbau eines Computersystems [1, S. 18]
- Hauptkomponenten eines Computers sind Prozessor, Speicher und ein E/A-Subsystem
- Busse → Verbindungen dazwischen



Bussysteme und Arbeitsspeicher

Bussystem

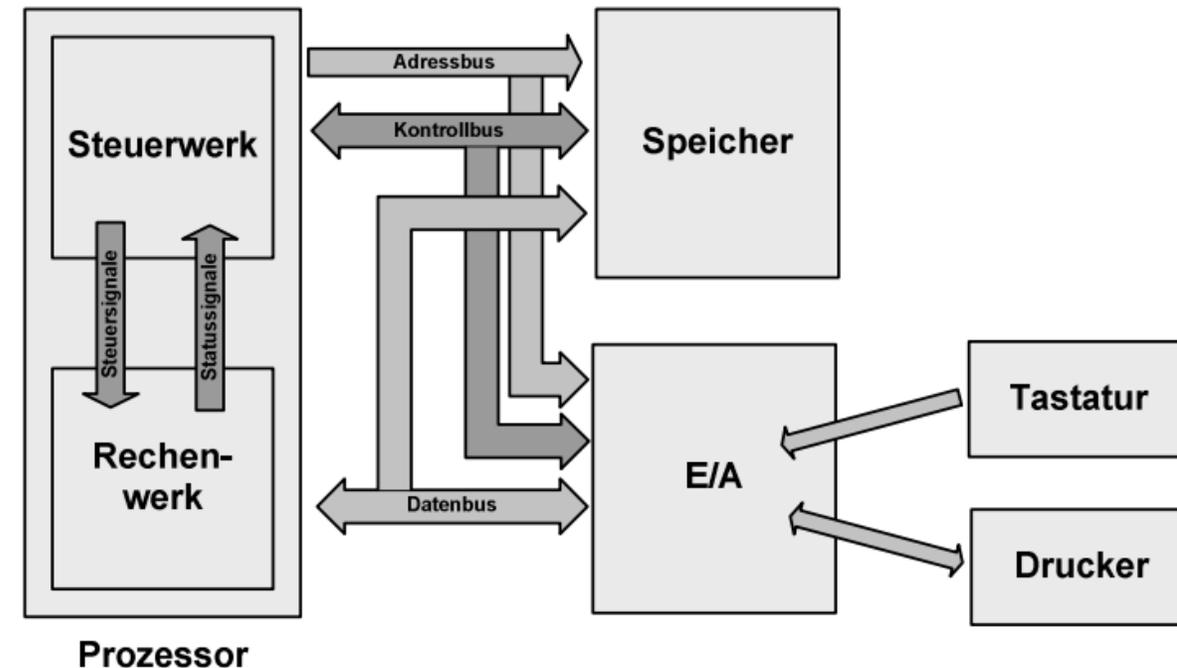
- Leitungsbündel zur Datenübertragung zwischen den Einheiten des Rechners
- Speicherzugriff und die Datenübermittlung zwischen Speicher und Prozessor erfolgen über einen gemeinsamen Weg - den Datenbus (durchgängigen Adressierung)
- So entsteht hier eine Engstelle, der sogenannte „Flaschenhals“.
- Drei Teilbusse: Datenbus, Adressbus, Steuerbus



Bussysteme und Arbeitsspeicher^[1]

Bussystem

- **Transport der Daten und Programmbefehle vom Arbeitsspeicher zum Prozessor**
- Steuerbus: bestimmt, ob Informationen gelesen oder geschrieben werden sollen
- Adressbus: übernimmt die Adresse einer Speicherzelle oder eines Ein- oder Ausgabegerätes, in der die Information abgelegt oder von der aus gelesen werden soll
- Datenbus: überträgt Daten zwischen den Teilsystemen des Prozessors und dem Arbeitsspeicher



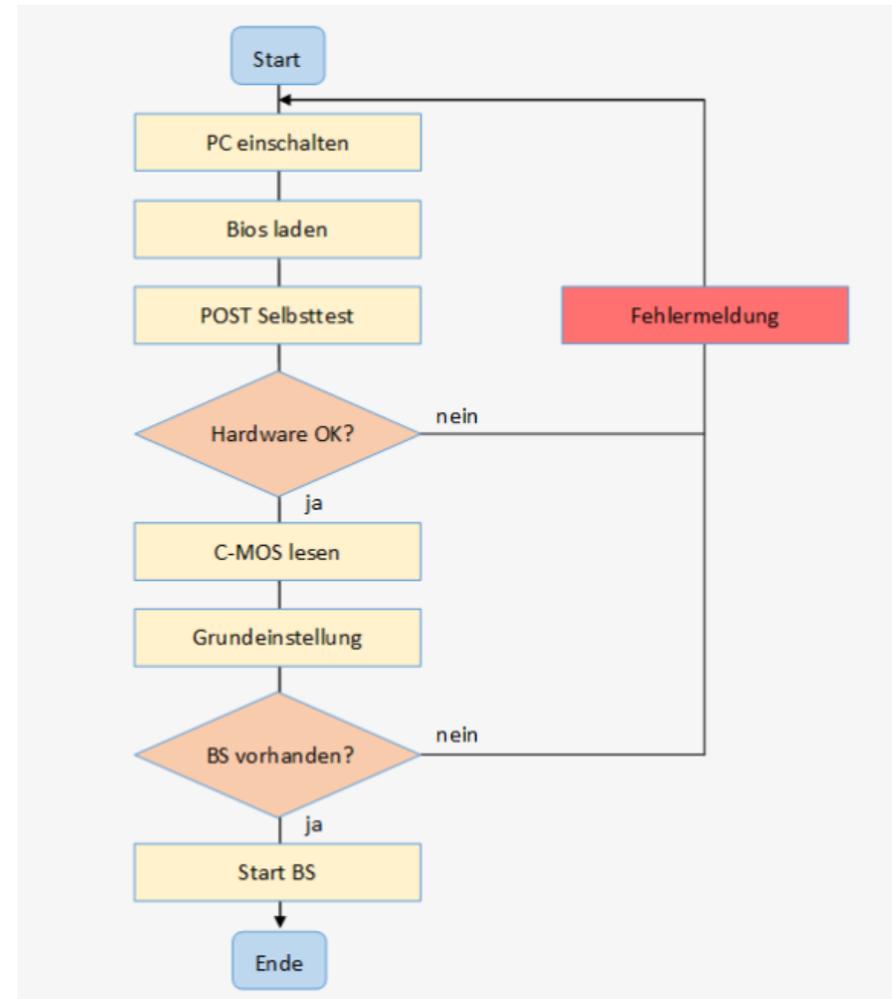
Bussysteme und Arbeitsspeicher^[1]

Arbeitsspeicher

- Der Arbeitsspeicher ist deutlich kleiner als eine Festplatte oder ein Magnetband und verliert seine Informationen beim Ausschalten des Rechners.
- schneller als permanente Speichermedien
- besteht aus Halbleiter-Speicherbausteinen, RAM genannt
- Zu startende Programme werden vom Prozessor zunächst automatisch von der Festplatte oder einem anderen Medium dorthin kopiert
- Zwischenspeicher für alle Daten, die während des Programmlaufes erzeugt oder verarbeitet werden

Ablauf des Bootvorgangs

- Mehrstufiger, aufeinander aufbauender Vorgang → macht Computer arbeitsfähig
- Firmware (BIOS oder UEFI)
- Schritt 1: Aufruf des BIOS (Basic Input Output System)
- Bios stellt die klassische Variante dar
- Extensible Firmware Interface (EFI), bzw. Unified Extensible Firmware Interface (UEFI) als Nachfolger



BIOS

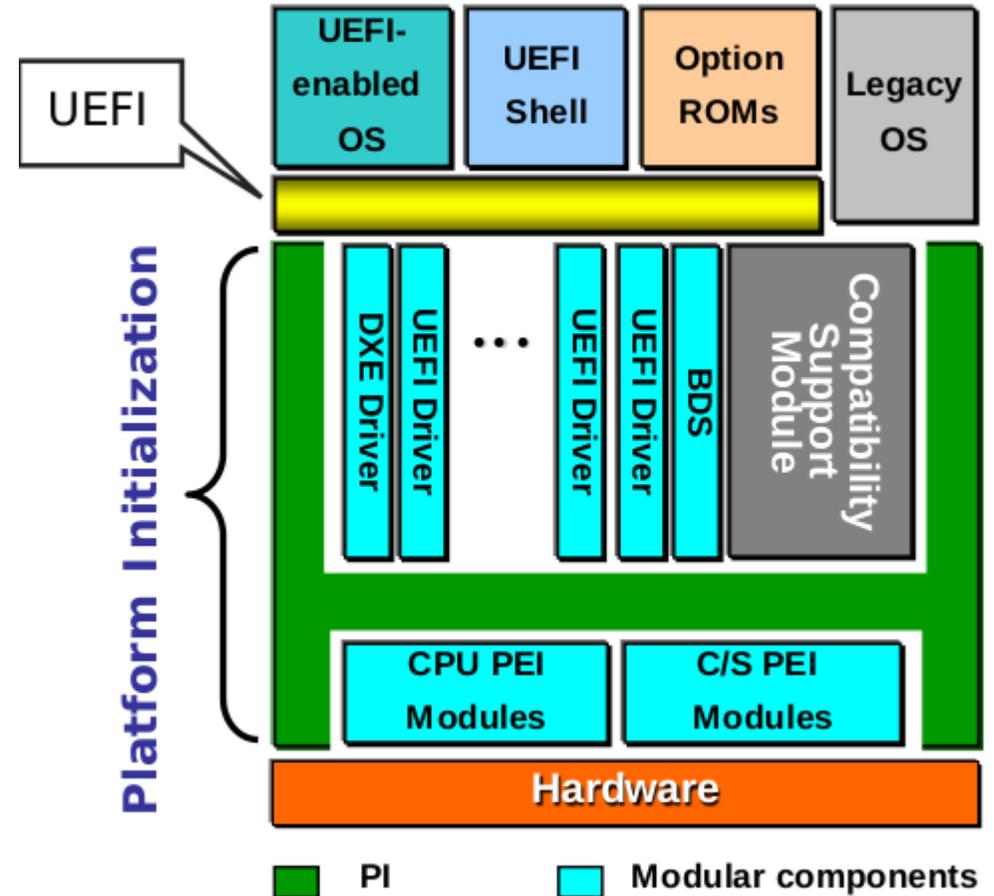
- Computercode, der sich in einem nicht-volatilen Speicher auf dem Mainboard des Rechners befindet
- unmittelbar nach dem Start des Computers ausgeführt
- Arbeitet zwischen Hardware und Betriebssystem (Schnittstelle zwischen Soft- und Hardware)
- Führt Hardwareerkennung durch, initialisiert diese
 - Testen der Komponenten CPU, RAM, ggf. Grafikkarte) → Power-On self Test (POST)
 - Einbinden des Hardwarespeichers
 - Initialisierung von Hardwareeinstellungen

BIOS

- Lädt Einstellungen (Bootreihenfolge, Partitionstabelle und Master Boot Record)
- MBR umfasst Bootprogramm, welches Betriebssystemeinträge und Partitionseinträge enthält
- BIOS übergibt Vorgang an Bootmanager → Manager sucht und lädt Kernel → Übergabe an Kernel

UEFI

- mehr Möglichkeiten als BIOS
- Firmware, die eigene Treiber zum Einbinden von Festplatten oder Netzwerkkarten besitzt
- Kann Partitionstabellen lesen, versteht FAT-Systeme



Datenträgertechnik

Speichertechnologien,

Partitionierung

MBR

GUID

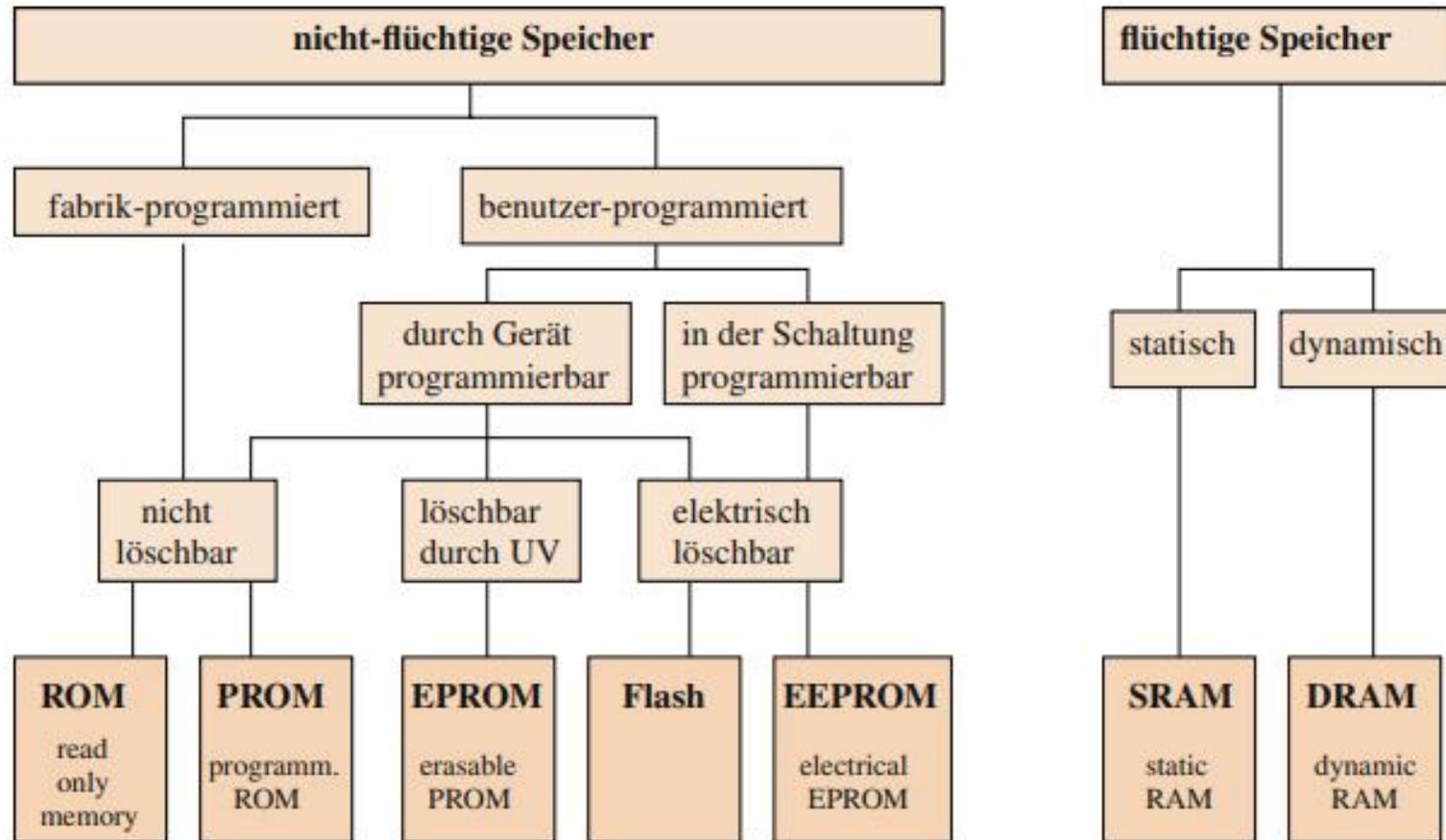
Speichertypen

Ein Festplattenlaufwerk oder auch Hard-Drive ist ein magnetisches Speichermedium. Bei einer Festplatte werden die Daten auf rotierenden Scheiben abgelegt. Dazu Plattenoberfläche mit Hilfe eines Schreib-/Lesekopfes magnetisiert. Durch die Remanenz erfolgt die Speicherung der Information. Das Auslesen der Information erfolgt durch Abtastung der Magnetisierung der Plattenoberfläche. Weitere Datenträger, die mittels Magnetisierung Informationen ablegen sind.

Neben klassischen Festplatten-Laufwerken werden zunehmend so genannte Solid-State-Drives (SSDs) in IT-Systemen verbaut. Ein SSD, ist ein nichtflüchtiges elektronisches Speichermedium. Vorteile einer SSD gegenüber herkömmlichen Laufwerken sind mechanische Robustheit, sehr kurze Zugriffszeiten und keine Geräuschentwicklung aufgrund beweglicher Bauteile.

USB-Sticks, Memory-Cards in Mobil-Telefonen, SD-Karten nutzen genauso wie die zuvor genannten SSDs Flash-Speicher zur Speicherung. Bei einem Flash-EEPROM-Speicher wird Information (Bits) in einer Speichereinheit (Speicherzelle) in Form von elektrischen Ladungen gespeichert. Alle Flash Speicher haben eine begrenzte Lebensdauer, die in einer maximalen Anzahl an Löschzyklen angegeben wird. Diese schwanken zwischen 10.000 und hin zu zwei Millionen Zyklen.

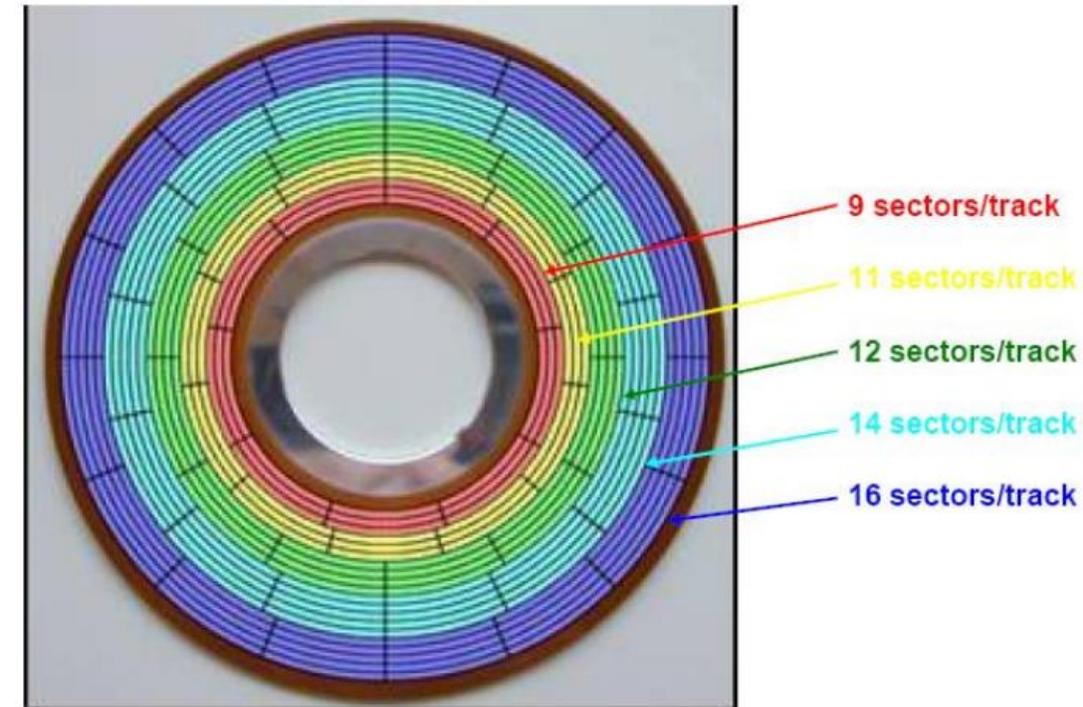
Übersicht über Speichertechnologien [2]



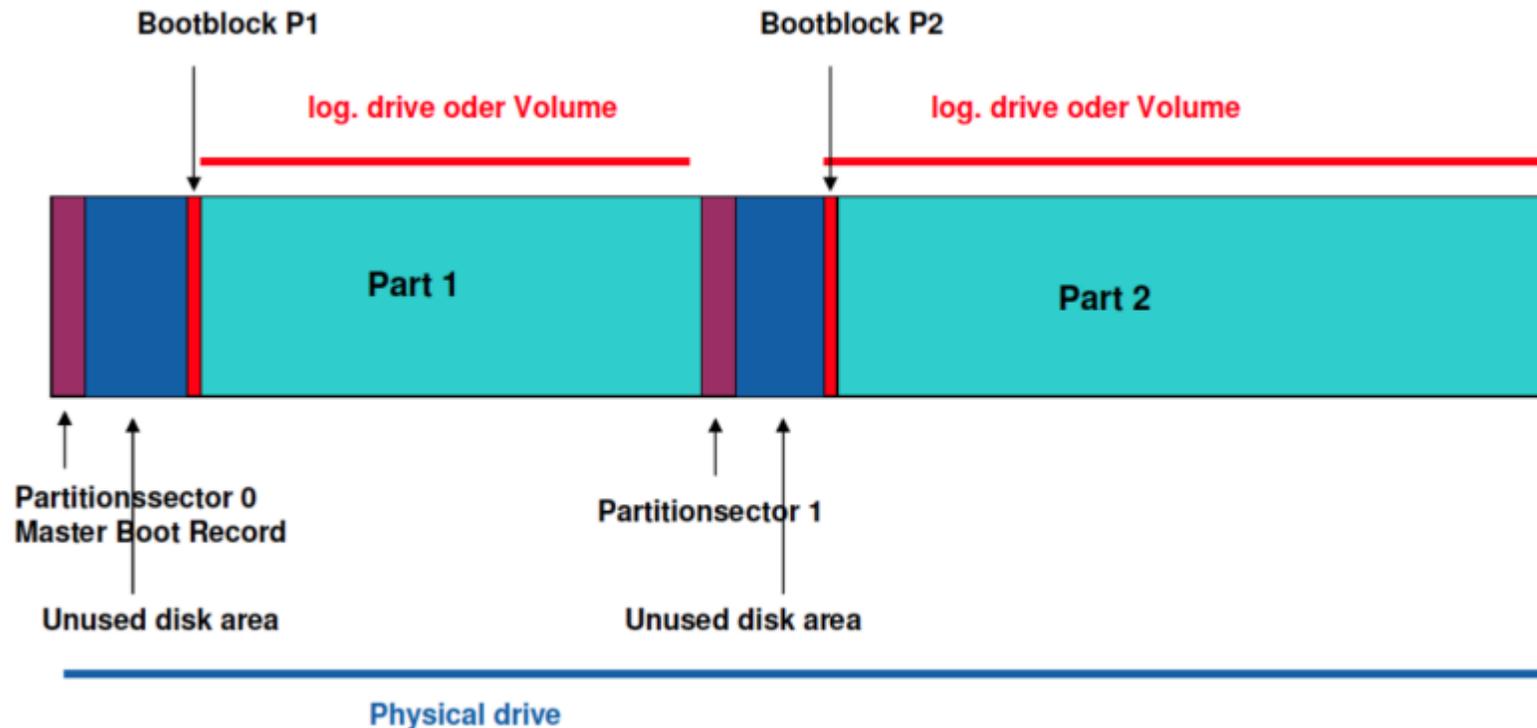
Technischer Aufbau von Festplatten

Moderne Festplatten: Zone Bit Recording

- Äußere Spuren wesentlich länger, bei gleicher Datendichte mehr Daten speicherbar
- Einteilung der Oberfläche in mehrere Zonen, äußere Zonen mit mehr Sektoren pro Spur
- Folge: höhere Datenrate außen
- Nummerierung Zylinder von außen nach innen
- schnelle Bereiche werden zuerst genutzt
- Innerste Spur für Parkposition („Autopark“)

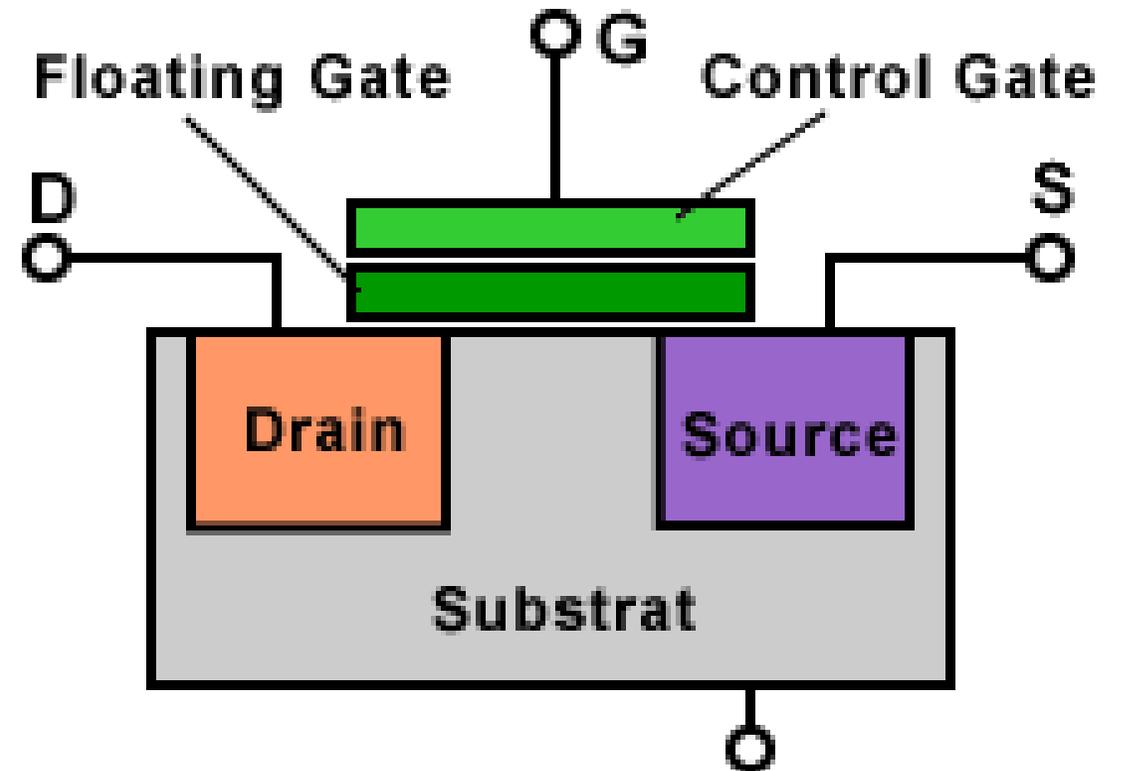


Technischer Aufbau von Festplatten



Technischer Aufbau von Flashspeichern

- Mehrere Speicherzellen jeweils mit Transistor zur Speicherung
- Speicherkarten, USB-Sticks, SSDs (Solid State Disks) und als Speicher für das BIOS von Mainboards
- elektrisch gelöscht und beschrieben



Technischer Aufbau von Flashspeichern

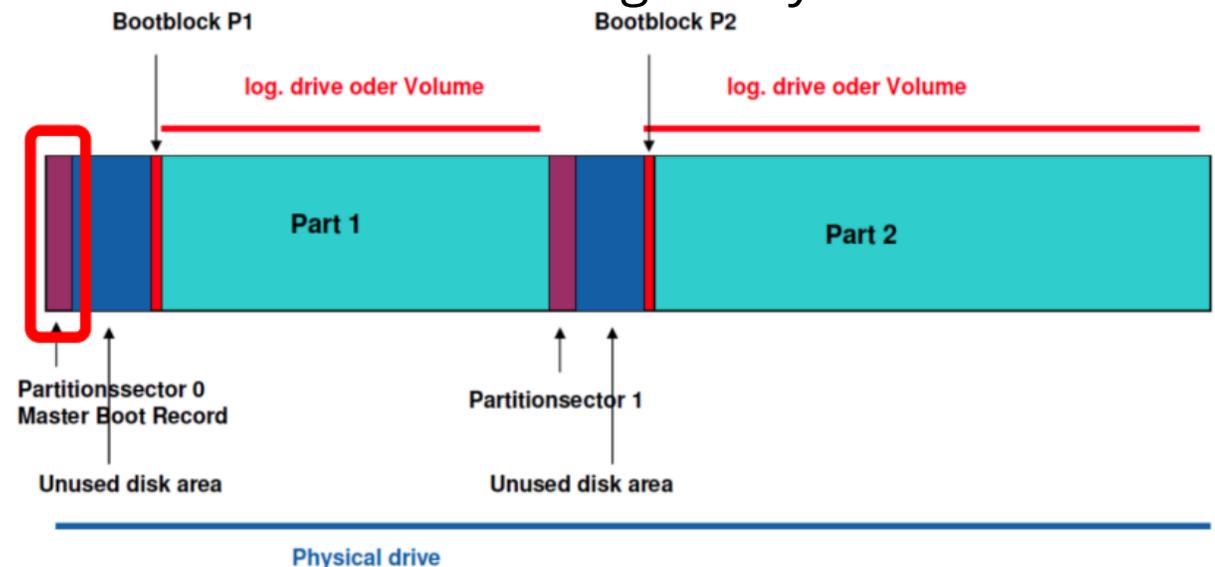
- Solid State Drive = „Flash-only“ Speicher
 - Single-level cell (SLC) vs. Multi-level cell (MLC)
 - NAND vs. NOR Zellen
- Vorteile:
 - Keine mechanischen Teile (günstig & robust)
 - Schnell (Datenrate & Latenz)
 - Kapazität „nach oben offen“
 - Strombedarf / Thermisches Verhalten
 - (Dynamic) Wear Levelling
 - Kein Defragmentieren notwendig

Technischer Aufbau von Flashspeichern

- Nachteile
 - Controller = SPOF (Single Point of Failure)
 - Recovery extrem aufwendig bis unmöglich
 - „sicheres Löschen“ nur im Enterprise Sektor
 - Leistungsverluste (Trim / Garbage Collection)

Grundlagen der Partitionierung

- Sektor: kleinste in einem Zugriff les- und schreibbare Einheit
- Partition: Unterteilung des Speicherplatzes eines Datenträgers in logische Bereiche
- Anwendung:
 - Exklusive Zuweisung von Speicherbereichen für Anwendungen / Systemen
 - Daten- / System-Sicherheit
 - Backup / Recovery
 - Multi-Boot Systeme



Master Boot Record (MBR)

- Die ersten 512 Bit einer Festplatte (erster Sektor)
- Für den Boot- und Startvorgang externe Speichermedien
- Informationsquelle:
 - Bootloader/Startprogramm
 - Datenträgersignatur
 - Partitionstabelle
 - Bootsektorsignatur

| Adresse | | Funktion / Inhalt | Größe (Bytes) |
|---------|-----|---|---|
| hex | dez | | |
| 0x0000 | 0 | Startprogramm (Master Boot Code / Bootloader) | 440 |
| 0x01B8 | 440 | Datenträgersignatur | 4 |
| 0x01BC | 444 | Null (0x0000) | 2 |
| 0x01BE | 446 | Partitionstabelle | 64 |
| 0x01FE | 510 | 55 _{hex} | Bootsektor-Signatur (wird vom BIOS für den ersten Bootloader geprüft) |
| 0x01FF | 511 | AA _{hex} | |

Master Boot Record (MBR)

Partitionstabelle

- 4 Einträge mit je 16 Bit
- Fest spezifiziert

| Offset | Länge | Beschreibung |
|--------|-------|---|
| 0x00 | 1 | Bootindikator (0x80 für bootfähig) |
| 0x01 | 3 | Startsektor der Partition in CHS Notation |
| 0x04 | 1 | Partitionstyp |
| 0x05 | 3 | Partitionsende in CHS Notation |
| 0x08 | 4 | Startsektor der Partition in LBA Notation |
| 0x0C | 4 | Größe der Partition in Sektoren |

Master Boot Record (MBR)

Partitionstabelle (Partitionstypen)

Command (m for help): l

| | | | | | | | |
|----|-----------------|----|-----------------|----|-----------------|----|-----------------|
| 0 | Empty | 1c | Hidden Win95 FA | 70 | DiskSecure Mult | bb | Boot Wizard hid |
| 1 | FAT12 | 1e | Hidden Win95 FA | 75 | PC/IX | be | Solaris boot |
| 2 | XENIX root | 24 | NEC DOS | 80 | Old Minix | c1 | DRDOS/sec (FAT- |
| 3 | XENIX usr | 39 | Plan 9 | 81 | Minix / old Lin | c4 | DRDOS/sec (FAT- |
| 4 | FAT16 <32M | 3c | PartitionMagic | 82 | Linux swap | c6 | DRDOS/sec (FAT- |
| 5 | Extended | 40 | Unix 80286 | 83 | Linux | c7 | Syrinx |
| 6 | FAT16 | 41 | PPC PReP Boot | 84 | OS/2 hidden C: | da | Non-FS data |
| 7 | HPFS/NTFS | 42 | SFS | 85 | Linux extended | db | CP/M / CTOS / . |
| 8 | AIX | 4d | QNX4.x | 86 | NTFS volume set | de | Dell Utility |
| 9 | AIX bootable | 4e | QNX4.x 2nd part | 87 | NTFS volume set | df | BootIt |
| a | OS/2 Boot Manag | 4f | QNX4.x 3rd part | 8e | Linux LVM | e1 | DOS access |
| b | Win95 FAT32 | 50 | OnTrack DM | 93 | Amoeba | e3 | DOS R/O |
| c | Win95 FAT32 (LB | 51 | OnTrack DM6 Aux | 94 | Amoeba BBT | e4 | SpeedStor |
| e | Win95 FAT16 (LB | 52 | CP/M | 9f | BSD/OS | eb | BeOS fs |
| f | Win95 Ext'd (LB | 53 | OnTrack DM6 Aux | a0 | IBM Thinkpad hi | ee | EFI GPT |
| 10 | OPUS | 54 | OnTrackDM6 | a5 | FreeBSD | ef | EFI (FAT-12/16/ |
| 11 | Hidden FAT12 | 55 | EZ-Drive | a6 | OpenBSD | f0 | Linux/PA-RISC b |
| 12 | Compaq diagnost | 56 | Golden Bow | a7 | NeXTSTEP | f1 | SpeedStor |
| 14 | Hidden FAT16 <3 | 5c | Priam Edisk | a8 | Darwin UFS | f4 | SpeedStor |
| 16 | Hidden FAT16 | 61 | SpeedStor | a9 | NetBSD | f2 | DOS secondary |
| 17 | Hidden HPFS/NTF | 63 | GNU HURD or Sys | ab | Darwin boot | fd | Linux raid auto |
| 18 | AST SmartSleep | 64 | Novell Netware | b7 | BSDI fs | fe | LANstep |
| 1b | Hidden Win95 FA | 65 | Novell Netware | b8 | BSDI swap | ff | BBT |

→ Screenshot fdisk

Master Boot Record (MBR)

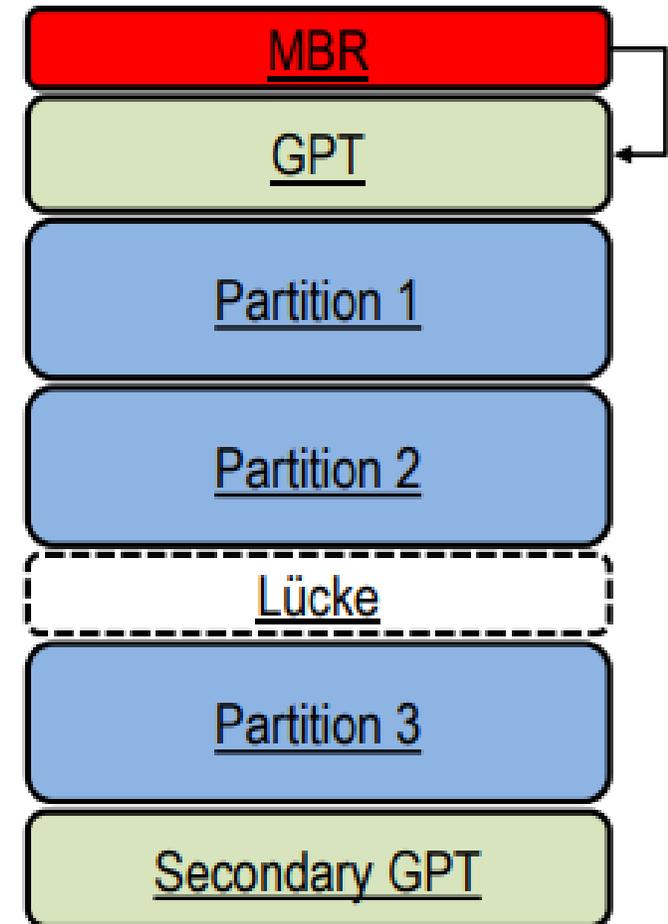
MBR eines FAT32-formatierten Speichermediums

```
00000190: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000001A0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000001B0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 80 01 .....
000001C0: 01 00 0B FE 3F 65 3F 00 - 00 00 A7 00 19 00 00 00  .o.??e?...|.
000001D0: 01 66 05 FE BF 10 E 00 - 19 00 FB AB 68 00 00 00  .f+?µ.|.0%h..
000001E0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000001F0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 55 AA .....U
Sector 0 of 8.06.781 Hard Disk 2 absolute Sector 0
offset 477, hex 1DD
```

- 3.: 1 Byte Dateisystem
0B: FAT32
- 4.: 3 Byte Ende CHS Partition
- 5.: 4 Byte Startsektor relativ zum Anfang der Platte
- 6.: 4 Byte Länge der Partition in Sektoren
- 1.: 1 Byte Bootindikator
(0x80 aktiv, 0x00 inaktiv)
- 2.: 3 Byte Start CHS Partition

GUID Partition Table (GPT)

- Global Unique Identifier Partition Tabel
- Löst Probleme des MBR (begrenzte Größe, Partitionsanzahl)
- Enthält Sicherungen der Partitionstabellen (verbesserte Integrität)



GUID Partition Table (GPT)

| Offset | Länge | Beschreibung |
|--------|-------|---|
| 0x0 | 8 | Signatur immer „EFI PART“ |
| 0x8 | 4 | Revision, derzeit V.1.0 => 00 00 01 00 |
| 0x0C | 4 | Headersize |
| 0x10 | 4 | CRC 32 Checksumme |
| 0x14 | 4 | Reserviert – immer 0 |
| 0x18 | 8 | Primary LBA, Adresse des primary GPT headers |
| 0x20 | 8 | Backup LBA, Adresse des backup GPT headers |
| 0x28 | 8 | 1. Benutzbare LBA Sektor |
| 0x30 | 8 | Letzter benutzbarer LBA Sektor |
| 0x38 | 16 | Disk GUID, ID über die Disk und die Partitionstabelle |
| 0x48 | 8 | Partition entry LBA, Beginn der Partitionseinträge |
| 0x50 | 4 | Anzahl der Partitionseinträge (bei Windows 128) |
| 0x54 | 4 | Größe der einzelnen Partitionseinträge (128 Bytes) |
| 0x58 | 4 | CRC 32 Checksumme über die Partitionseinträge |

RAID-Systeme

RAID

- RAID = Redundant Array of Independent Disk
- Methode zur Speicherung von Daten auf Festplatten
- Schutz vor Hardwareausfall durch Redundanzen
- Zusammenfassen von Festplatten zu Gruppen die unter anderem Datenkopien beinhalten
- Techniken: Mirroring, Striping, Parität

RAID

Hardware-RAID

- Arbeit wird hierbei vom Raid-Controller übernommen
- CPU wird nicht durch Berechnungen des RAIDs belastet

Vorteile:

- Steht bereits beim Booten zur Verfügung
- Unterstützt eine Vielzahl von Betriebssystemen
- Hohe Performance
- Niedrige CPU-Last am Host

Nachteile:

- Hohe Anschaffungskosten
- Betriebssystemabhängig



Software-RAID

- Kein separater RAID-Controller benötigt
- Preisgünstiger als ein Hardware-RAID
- Kann unter anderem direkt unter Windows erstellt werden

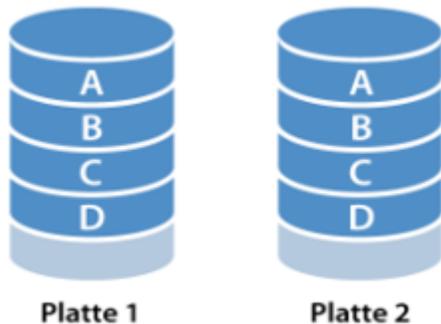
Nachteile:

- schlechtere Gesamtleistung durch höherer CPU-Last

RAID – Techniken

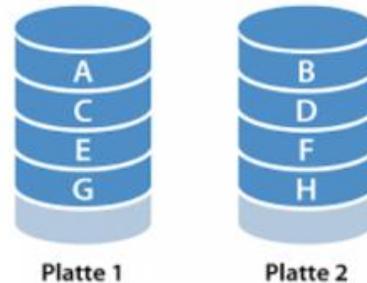
Mirroring

schreibt Daten gleichzeitig auf zwei Laufwerke



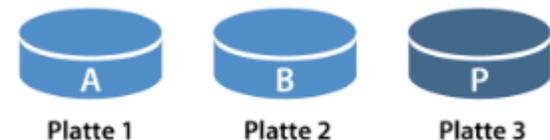
Striping

Daten auf die verfügbaren Laufwerke in Blöcken (Streifen) verteilt



Parität

Ist eine Summenverknüpfung, mit der man verlorene Daten wiederherstellen kann (und Kontrolle bei Veränderung)

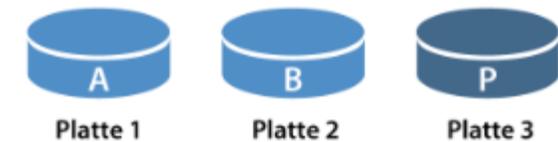


Je nach RAID-Level werden die Techniken kombiniert!

RAID – Techniken

Parität

- Möglichkeit verlorene Daten wiederherzustellen
- Datensatz AB: fällt ein Teil aus, lässt er sich mit dem übrigen Teil und dem Paritätsblock wiederherstellen
- Parität mittels Summenverknüpfung



Bits der Datenplatten → Summe → gerade oder ungerade → Summen-Bit der Paritätsplatte

```
0 + 0 + 0 → Summe ist 0 → Summe ist gerade → Summen-Bit 0
1 + 0 + 0 → Summe ist 1 → Summe ist ungerade → Summen-Bit 1
1 + 1 + 0 → Summe ist 2 → Summe ist gerade → Summen-Bit 0
1 + 1 + 1 → Summe ist 3 → Summe ist ungerade → Summen-Bit 1
0 + 1 + 0 → Summe ist 1 → Summe ist ungerade → Summen-Bit 1
```

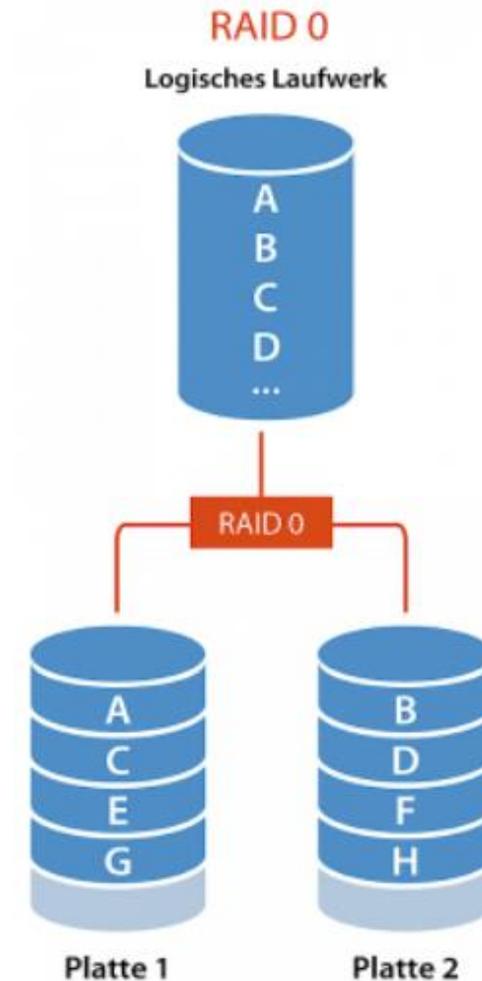
Wenn beispielsweise das Bit der ersten Datenplatte verloren ginge, könnte man es aus den Bits der anderen Datenplatten und dem Summen-Bit der Paritätsplatte errechnen.

```
? + 0 + 0 → eine gerade Summe → das erste Bit kann nicht 1 sein, nur 0
? + 0 + 0 → eine ungerade Summe → das erste Bit kann nicht 0 sein, nur 1
? + 1 + 0 → eine gerade Summe → das erste Bit kann nicht 0 sein, nur 1
? + 1 + 1 → eine ungerade Summe → das erste Bit kann nicht 0 sein, nur 1
? + 1 + 0 → eine ungerade Summe → das erste Bit kann nicht 1 sein, nur 0
```

RAID- Level

RAID 0

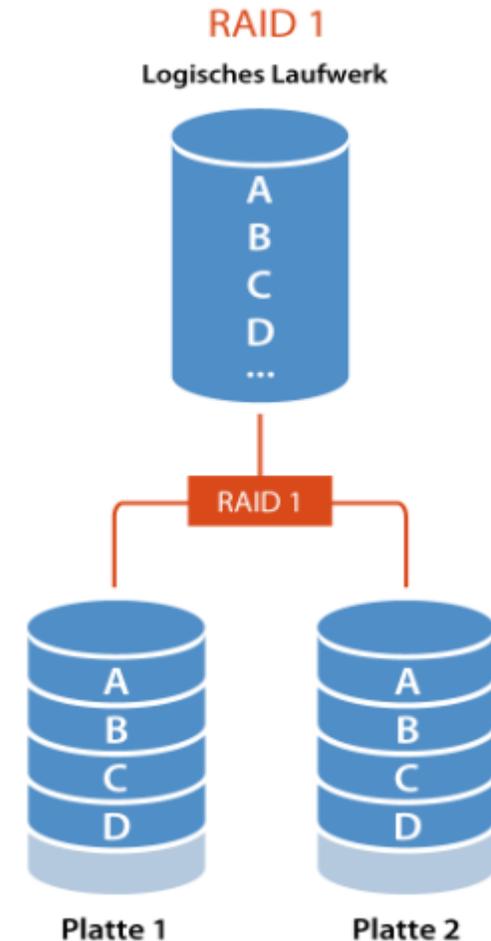
- Mindestens 2 Festplatten
- Daten werden auf Festplatten verteilt
- Keine Sicherheit (fällt eine Platte aus, sind alle Daten weg)
- Hohe Lese- und Schreibgeschwindigkeit
- Speicherkapazität 100 %



RAID- Level

RAID 1

- Mindestens 2 Festplatten
- Alle Daten werden auf beide Festplatten geschrieben
- Hohe Ausfallsicherheit
- Hohe Lese- und Schreibgeschwindigkeit
- Speicherkapazität 50 %



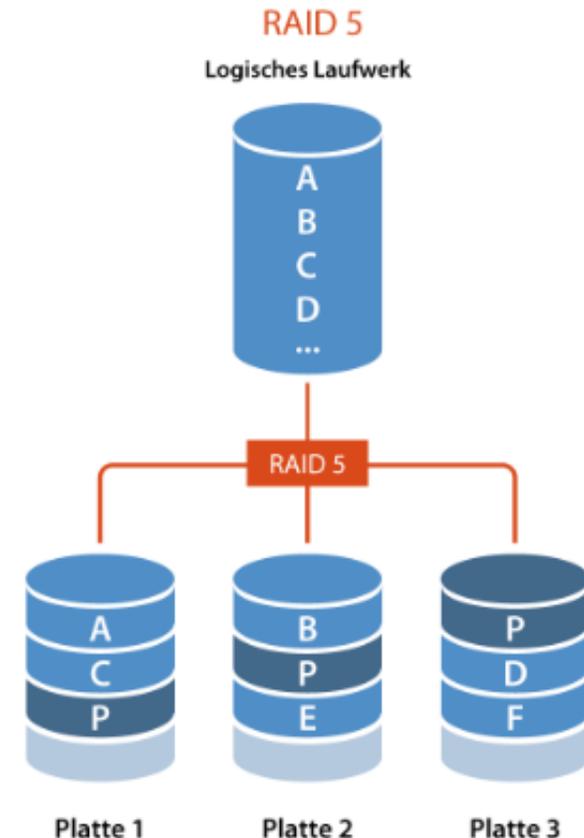
RAID- Level

RAID 5

- Mindestens 3 Festplatten (max. 16)
- Daten auf Festplatten verteilt, Parität errechnet und gespeichert
- Bei Festplattenausfall können über Parität Daten gerettet werden
- Hohe Rechenleistung, aber Ausfallsicherheit
- Speicherkapazität: 67 % - 94 %

RAID 3/4: Paritäten auf einer Platte

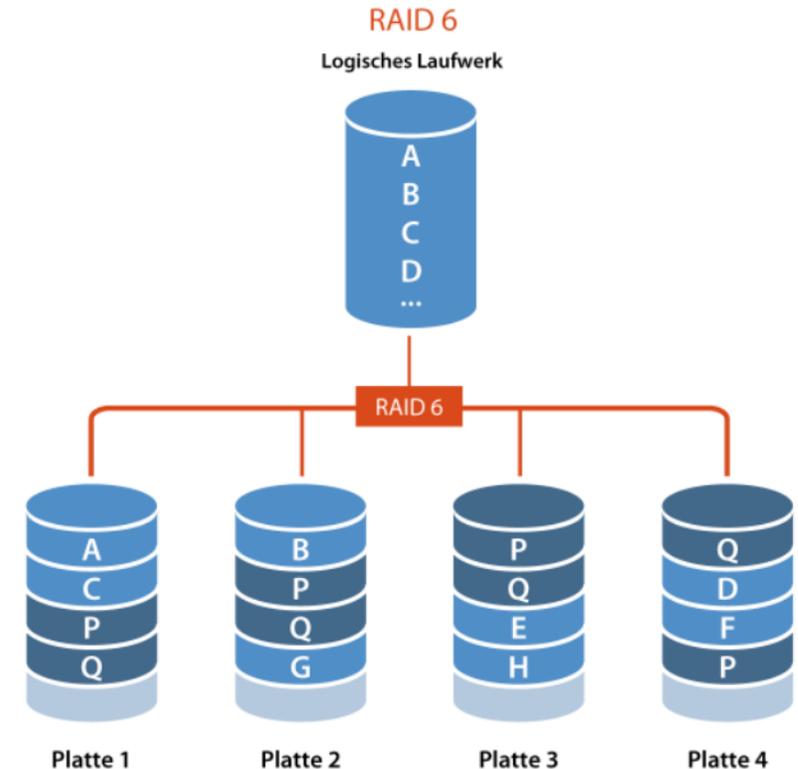
RAID 5: Paritäten verteilt über alle Platten



RAID- Level

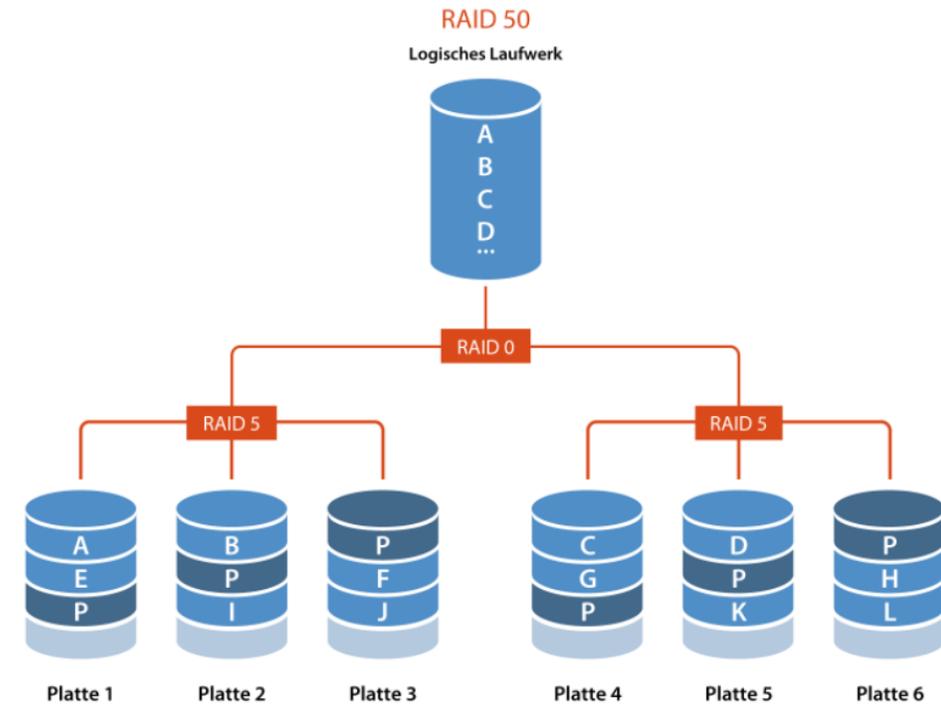
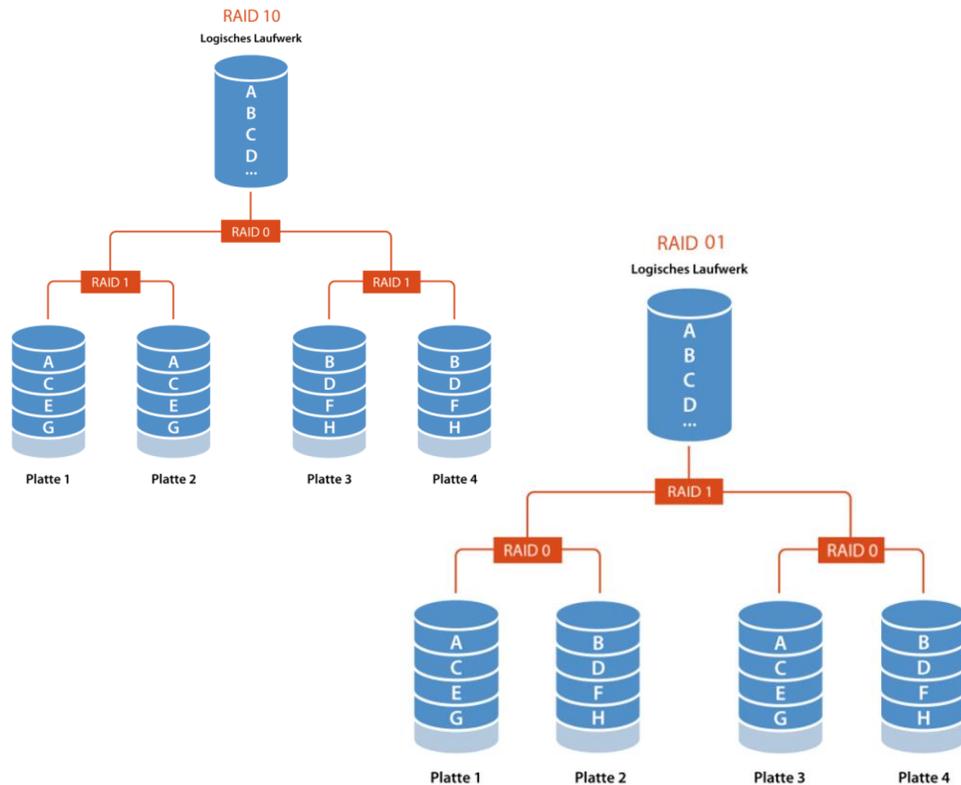
RAID 6

- Mindestens 4 Festplatten (max. 16)
- Daten auf Festplatten verteilt, zwei Paritäten (P und Q) errechnet und gespeichert
- Bei Festplattenausfall (maximal 2) können über Parität Daten gerettet werden
- Extrem hohe Rechenleistung, aber hohe Ausfallsicherheit
- Speicherkapazität: 50 % - 88 %



RAID – Levelkombinationen

- Kombination verschiedener Level zur Leistungs- oder Ausfallsicherungsoptimierung



Literatur

[1] Hellmann, Roland: Rechnerarchitektur – Einführung in den Aufbau moderner Computer. 2. Auflage. 2016.

[2] Häberlein, Tobias. "Speicher." Technische Informatik. Vieweg+ Teubner, 2011. 181-196.

Vielen Dank



**HOCHSCHULE
MITTWEIDA**
University of Applied Sciences

Prof. Dr. rer. nat. Dirk Labudde

Hochschule Mittweida | University of Applied Sciences
Technikumplatz 17 | 09648 Mittweida
Fakultät Computer- und Biowissenschaften | Fraunhofer Lernlabor

T +49 (0) 3727 58-1469

F +49 (0) 3727 58-21469

labudde@hs-mittweida.de

Haus 8 | Richard Stücklen-Bau | Raum 8-105
Am Schwanenteich 6b | 09648 Mittweida

[hs-mittweida.de](https://www.hs-mittweida.de)